CSE 341 Final — June 8, 2000 — Answer Key

This exam is open book, open notes, 110 minutes, 110 points total.

1. (18 points) Consider the following program. This program will print out six values.

```
begin
integer n;
procedure p(j: integer)
    begin
    n := 100;
    j := j+10;
    print(n);
    print(j);
    end;
n := 0;
p(n);  print(n);
p(n);  print(n);
end;
```

   (a) What is the output if the parameters are passed by value?
   100 10 100
   100 110 100

   (b) What is the output if the parameters are passed by value-result?
   100 10 10
   100 20 20

   (c) What is the output if the parameters are passed by reference?
   110 110 110
   110 110 110

2. (5 points) A static method in Java may not use the `this` reference. Why not?

   A static method can be invoked on the class that holds the method, as well as on an instance of that class. If the method is invoked on the class, there won't be an instance created, and so the `this` reference would not be meaningful, since it refers to the instance on which the method was invoked.

3. (27 points) Suppose that the following Miranda script has been filed in.

```
triple x = 3*x
add1 x = x+1

makebit False = 0
makebit True = 1

my_const k x = k

function_map [] x = []
function_map (f:fs) x = f x : function_map fs x

tree * ::= Leaf * | Node (tree *) (tree *)

height (Leaf x) = 0
|| max2 is a library function that finds the max of two numbers
height (Node left right) = 1 + max2 (height left) (height right)
```

```
treesum (Leaf x) = x
treesum (Node left right) = treesum left + treesum right

mytree = Node (Leaf 10) (Node (Leaf 20) (Leaf 30))
```

What is the result of evaluating the following Miranda expressions? If there is a compile-time type error, or a run-time error, or a non-terminating computation, say so. If the result is infinite, give the first several values. If the expression is followed by `::`, then give the type, instead of the value.

(a) `my_const ::`
`*->**->*`

(b) `function_map ::`
`[*->**]->*->[**]`

(c) `function_map [triple, makebit] ::`
`type error in expression`
`cannot cons num->num to [bool->num]`

(d) `function_map [triple,add1, my_const 100 ] 10`
`[30,11,100]`

(e) `function_map [my_const 100 ] ::`
`*->[num]`

(f) `function_map [my_const 100, makebit] ::`
`bool->[num]`

(g) `height ::`
`tree *->num`

(h) `treesum ::`
`tree num->num`

(i) `function_map [height,treesum] mytree`
`[2,60]`

4. (10 points) Consider the following definition of the `min` constraints in CLP($\mathcal{R}$).

```
min(X,Y,X) :- X<=Y.
min(X,Y,Y) :- X>Y.
```

Show the derivation tree for the following queries. (You can show either the full or simplified tree, or something in between.)

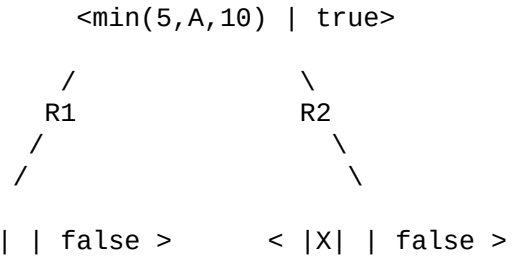
(a) `min(5,A,2).`
Here is a simplified derivation tree (where `|X|` is supposed to be a solid black box indicating a failed derivation):

```
R1: min(X,Y,X) :- X<=Y.
R2: min(X,Y,Y) :- X>Y.

          <min(5,A,2) | true>

         /              \
        R1              R2
       /                  \
      /                    \
                            _
< |X| | false >     < |_| | A=2 >
```

(b) `min(5,A,10).`



5. (12 points) Consider the following definition of the `min` and `sorted` constraints in CLP($\mathcal{R}$). (`min` is the same as in Question 4.)

```
min(X,Y,X) :- X<=Y.
min(X,Y,Y) :- X>Y.

sorted([]).
sorted([X]).
sorted([A,B|R]) :-
  A <= B,
  sorted([B|R]).
```

What answers are produced for the following queries? (Give all of the answers that would be produced by backtracking.)

(a) `min(A,B,10).`

```
A=10  10<=B ;

A>10  B=10
```

(b) `sorted([20,A,10]).`

```
no
```

(c) `sorted([A,B]), min(A,B,20).`

```
A=20  B>=20
```

(d) `sorted([A,B,C,D]), sorted([D,A,B,C]).`

```
A=B=C=D
```

6. (18 points) Consider the following Java class definition.

```
class PairOfStrings {

  public String a, b;

  public PairOfStrings(String a, b)
    {this.a = a; this.b = b;}

  public void flip()
    {String temp; temp = a; a = b; b = temp;}

  public boolean equals (PairOfStrings p)
    {return (a.equals(p.a) && b.equals(p.b));}

}
```

What is printed by the following code?

```
PairOfStrings p1 = new PairOfStrings("clam","squid");
PairOfStrings p2 = p1;
PairOfStrings p3;


System.out.println(p1 == p2);  ==> true

System.out.println(p1.equals(p2));  ==> true


p2.flip();

System.out.println(p1 == p2);  ==> true

System.out.println(p1.equals(p2));  ==> true


p3 = new PairOfStrings("squid","clam");

System.out.println(p1 == p3);  ==> false

System.out.println(p1.equals(p3)); ==> true
```

7. (10 points) Suppose that type checking (both at compile time and at run time) was eliminated entirely from Miranda. Are there any Miranda functions that currently don't compile due to a type error but that would run correctly in the version of the language with no type checking? If so, give an example. If not, explain why there aren't any such functions.

   Yes, there are Miranda functions that currently don't compile due to a type error but that would run correctly in the version of the language with no type checking. Here is an example.

   ```
   broken = const 3 ("octopus"/9)
   ```

   The definition for `broken` will give a compile-time type error, but would evaluate correctly without type checking. The second argument to `const` contains a type error (since an octopus is evenly divisible by 8 but not by 9), and so Miranda's type checker will complain. However, the library function `const` does not evaluate the second argument, and so if there were no type checking `broken` would just be bound to 3.

8. (10 points) Describe two significant similarities between Java and Smalltalk, and two significant differences.

   Two similarities are that both Java and Smalltalk are object-oriented languages, and that both use automatic storage management. Two differences are that in Smalltalk everything is an object, while in Java there are 8 primitive types whose values aren't objects (in addition to objects); and Java has type declarations and a static type system, while in Smalltalk there are no declarations and checking is done dynamically.